Engineering Science and Technology, an International Journal 19 (2016) 355–363

HOSTED BY

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: http://www.elsevier.com/locate/jestch

Full Length Article

Design, implementation and performance comparison of multiplier topologies in power-delay space

Mansi Jhamb, Garima *, Himanshu Lohani

University School of Information and Communication Technology, GGSIPU, Sector-16C, Dwarka, New Delhi, India

ARTICLE INFO

Article history:
Received 31 May 2015
Received in revised form
22 August 2015
Accepted 22 August 2015
Available online 11 September 2015

Keywords:
Double Pass Transistor (DPL) adders
Carry lookahead (CLA) adders
Domino CMOS logic
DPL Multiplier
CLA Multiplier

ABSTRACT

With the advancements in the semiconductor industry, designing a high performance processor is a prime concern. Multiplier is one of the most crucial parts in almost every digital signal processing applications. This paper addresses the implementation of an 8-bit multiplier design employing CMOS full adder, full adder using Double Pass Transistor (DPL) and multioutput carry Lookahead logic (CLA). DPL adder avoids the noise margin problem and speed degradation at low value of supply voltages associated with complementary pass transistor (CPL) logic circuits. Multioutput carry lookahead adder leads to significant improvement in the speed of the overall circuitry. The investigation is carried out with simulation runs on HSPICE environment using 90 nm process technology at 25 °C. Finally, the design guidelines are derived to select the most suitable topology for the desired applications. Investigation reveals that multiplier design using multioutput carry lookahead adder proves to be more speed efficient in comparison with the other two considered design strategies.

1. Introduction

The integrated circuit designers are facing new challenges due to the exponential growth in electronic devices and equipments in the past few years. Addition of more functionality along with real time applications demands revolutionary changes in the design process of a chip. Increase in computing requirements onto a single chip demands development of sophisticated tools that can perform complex operations which further results in increase in the processing power. Thus, development of high speed computational hardware i.e., adders and multipliers, is a prime concern in today's scenario. For low power and real time applications, computationally intensive digital signal processing algorithms are implemented in dedicated VLSI systems. Multiplier is one of the most crucial parts in such systems. The computation speed of processor is highly dependent on these arithmetic units. High performance processors result in increase in the complexity of the design. In order to improve the performance of processors there is a need to improve the complexity of such arithmetic units. Recent developments in portable electronic devices and digital signal processing systems demand flexible computational ability, low power utilization and shorter design cycles. Two most important design criteria deciding the performance of processor are speed and power consumption. In the literature many research efforts have been carried out in order to obtain energy efficient multiplier and adder architectures as demonstrated [1–7]. State-of-the-art designs focused mainly on reducing the silicon area but in the last decade the focus is primarily shifted toward speed and power. The complexity of the design directly depends on the speed of computation. High speed requirement results in increased complexity of the circuit, hence larger number of transistors will be required in the design which further results in high power dissipation. So there is a tradeoff between speed and power dissipation.

Adder is the basic component in any computational hardware. Thus, its performance characteristics directly affect the functioning of the entire system. Therefore, improvement in the performance of adder architecture is a prime concern as reported [8]. Various techniques can be employed externally or internally in order to improve the overall performance of any system. External techniques involve dealing with input data characteristics whereas internal techniques are concerned with the logic, circuit design and architecture of multiplier as reported [9–13]. The basic element of a multiplier design is the adder cell, which significantly affects the overall performance characteristics of a multiplier. The recent advancements in CMOS technology indicate a strong need for high speed, high density, low power, low cost multiplier design for

* Corresponding author. Tel.: +91 9873446811.
E-mail address: gs14.11.1990@gmail.com (Garima).
Peer review under responsibility of Karabuk University.

http://dx.doi.org/10.1016/j.jestch.2015.08.006

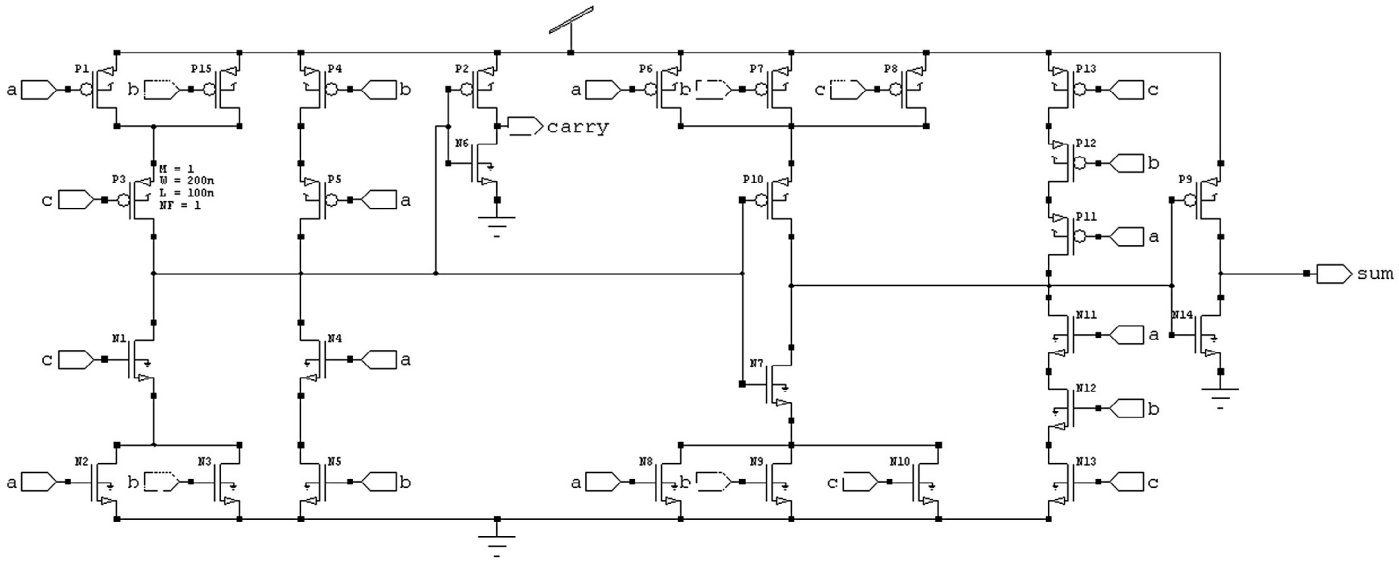

Fig. 1. Schematic of full adder.

ubiquitous use in majority of leading-edge commercial applications. In this paper an 8-bit Multiplier is designed with three different adder architectures i.e. CMOS full adder, Double Pass Transistor logic (DPL) as proposed by References 14 and 15 and multioutput Carry Lookahead (CLA) adder as demonstrated by Reference 16 and detailed analysis is carried out in terms of delay, power and power delay product (PDP). This paper proposes an 8-bit multiplier design using High speed multioutput CLA adders.

The remainder of this paper is organized as follows. In section 2, 8-bit adders are addressed using three different logic styles: CMOS full adder, DPL full adder and domino multioutput CLA adder architecture. In section 3, multiplier architectures are presented. Section 4 presents the comparison of the 8-bit full adders and 8-bit multipliers. Finally, in section 5, the conclusions are drawn.

2. Adder topologies analyzed

Adder is the basic architecture commonly used in every arithmetic circuit. In a multiplier design adders are used for partial product addition, and thus contribute toward the largest part of delay associated with whole multiplication process. This section presents different types of adders considered in this research.

2.1. CMOS full adder

CMOS full adder design is implemented using stack of PMOS and NMOS transistors. The basic architecture of full adder design consists of 28 transistors and three input variables (*a*, *b* and *c*) and two output variables (*sum* and *carry*) of 1-bit each. The schematic of full adder is shown in Fig. 1. The two outputs of the adder *sum* and *carry* are represented as:

$$\text{Sum} = a \oplus b \oplus c$$
$$\text{Carry} = a.b + b.c + c.a$$

The 8-bit ripple carry adder consists of eight full adder cells in cascade such that output carry of one full adder cell is applied as an input carry to another full adder cell. The architecture of an 8-bit ripple carry adder is shown in Fig. 2. Eight inputs a_7 to a_0 and b_7 to b_0 are applied to each of the full adder cell and output S_7 to S_0 represents eight bit sum from each full adder. The input carry of the first half adder cell must be grounded for the correct addition of least significant bits (a_0, b_0) otherwise it will result in erroneous output. As the whole computation solely relies on carry rippling, it results in large value of delay overhead. However ripple carry adder is one of the simplest adder architecture and requires less power consumption and area.

2.2. Double pass transistor logic (DPL) adder

Double pass transistor is a type of Pass transistor logic style. It is a modified version of complementary Pass transistor logic (CPL). CPL consists of only NMOS pass transistors, whereas the modified version DPL has both NMOS and PMOS transistors in symmetry in order to achieve full swing output from 0 to V_{DD} and reduced supply voltage.

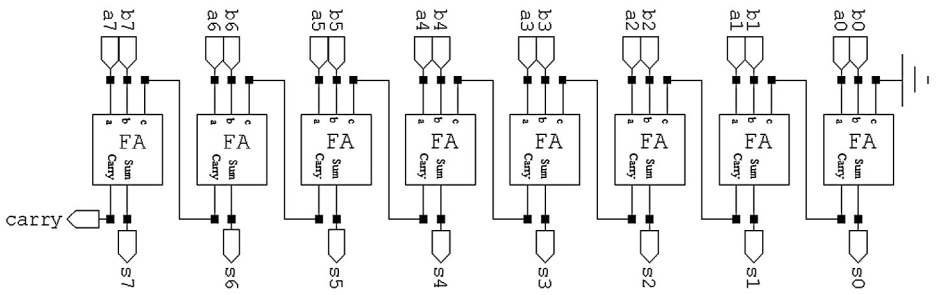

Fig. 2. Eight-bit Ripple Carry adder.

DPL adder compensates for the speed degradation of CMOS pass transistors in two ways as demonstrated [14,15]. It is a symmetrical arrangement in which any input is connected to the gate of one MOSFET and the source of another. Among the inputs any of A, A′, B, and B′ (A′ represents complement of A) is connected to the gates of NMOS transistor and to the sources of the NMOS and PMOS transistors. This results in balancing of input capacitance therefore, reducing the delay time dependency on the data. Second is the double transmission characteristic of DPL. This results in two outputs, one in normal form and another in complementary form.

The design of DPL Full adder is shown in Fig. 3. There are two inputs for each variable (*a, anot*), (*b, bnot*) and (*c, cnot*). Complementary outputs are *sum*, *sumnot* and *carry*, *carrynot*. The output *sum* of the full adder cell consists of XOR/XNOR logic gates, a multiplexer which has four inputs, two select lines *c* and *cnot*, two outputs complementary in nature, and an output buffer developed using CMOS logic. The output *carry* of the adder consists of AND/NAND logic gates, OR/NOR logic gates, a multiplexer, and a CMOS output buffer.

The 8-bit ripple carry adder using DPL is designed using eight DPL full adder cells in cascade. The complementary outputs can be grounded in parallel adder design to reduce the number of variables and to make design simple. The 8-bit parallel adder is same as shown in Fig. 2.

The advantage of DPL full adder as compared to CMOS full adder is its high speed. However, as it requires almost double transistors, the power dissipation is comparatively more in DPL.

2.3. Multioutput carry lookahead adder

In order to achieve high-speed in arithmetic operation, carry propagation time is the deciding factor as it limits the speed of the whole logic circuit which increases with the size of the adder. For n-bit Parallel adder,

$$\text{Total propagation delay} = S + (n-1)\cdot C$$

where S is propagation delay of sum and C is propagation delay in carry. In parallel adders (ripple carry adder), carry propagates in series or ripple which increases with size of adder.

Carry propagation time can be reduced by two ways as demonstrated [17]

- One solution is to develop faster gate with reduced delays.
- Another solution is to reduce the carry propagation delay at the cost of increasing the complexity of design. For reducing the carry propagation time in a parallel adder, several techniques are used out of which *Carry look-ahead adder logic* is most widely used.

Efstathiou et al. [16] have recently shown the 8-bit CLA adder using Manchester Carry Chain (MCC) in multioutput domino CMOS logic. The design includes two carry chains i.e. even and odd carry using series connected transistors shown in Fig. 4.

The MCC is the most commonly used domino CLA design as demonstrated [12]. The recursive properties of the carries have developed the multioutput domino gates. Other MCC adders in domino CMOS

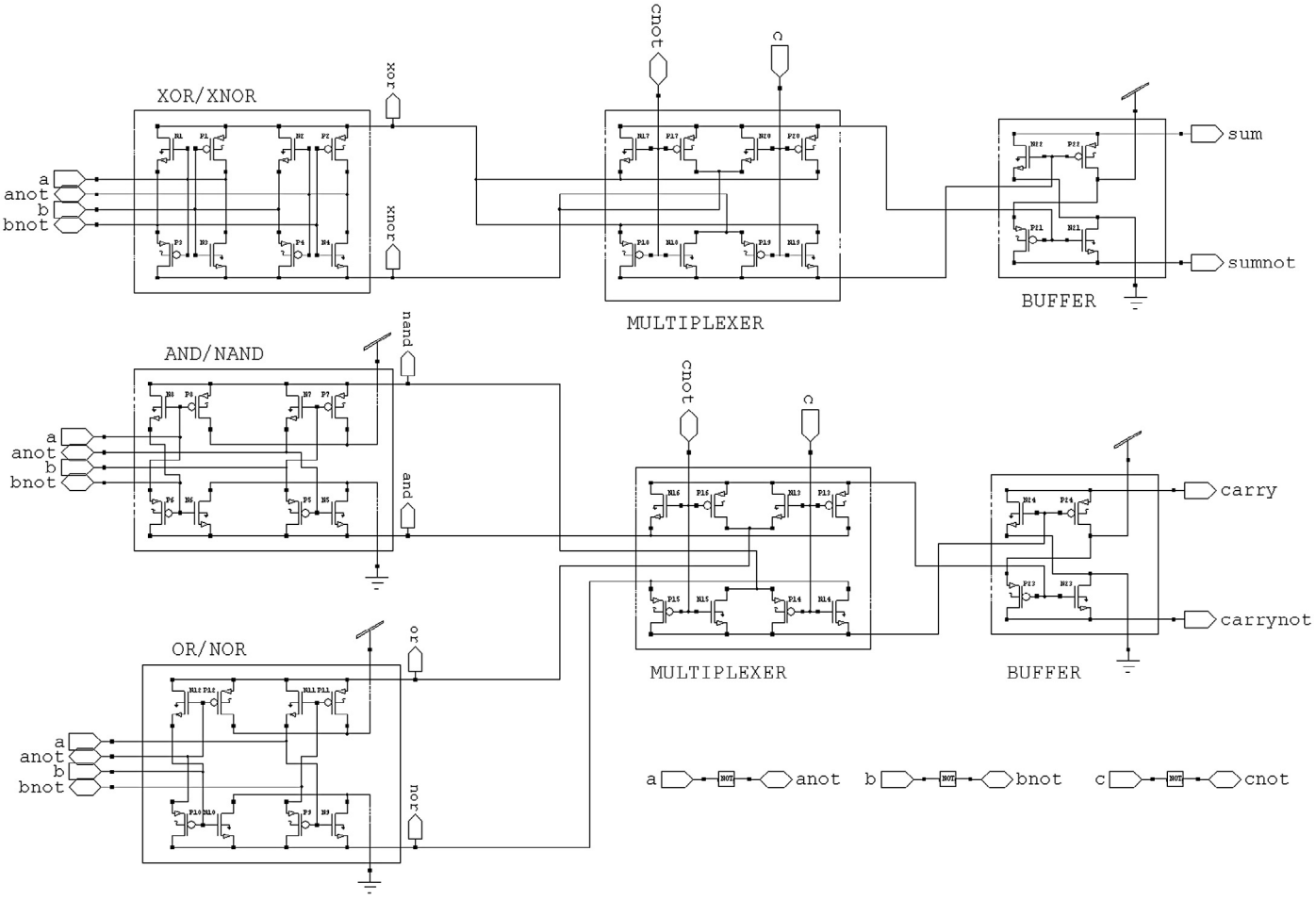

Fig. 3. DPL one-bit full adder cell [14].

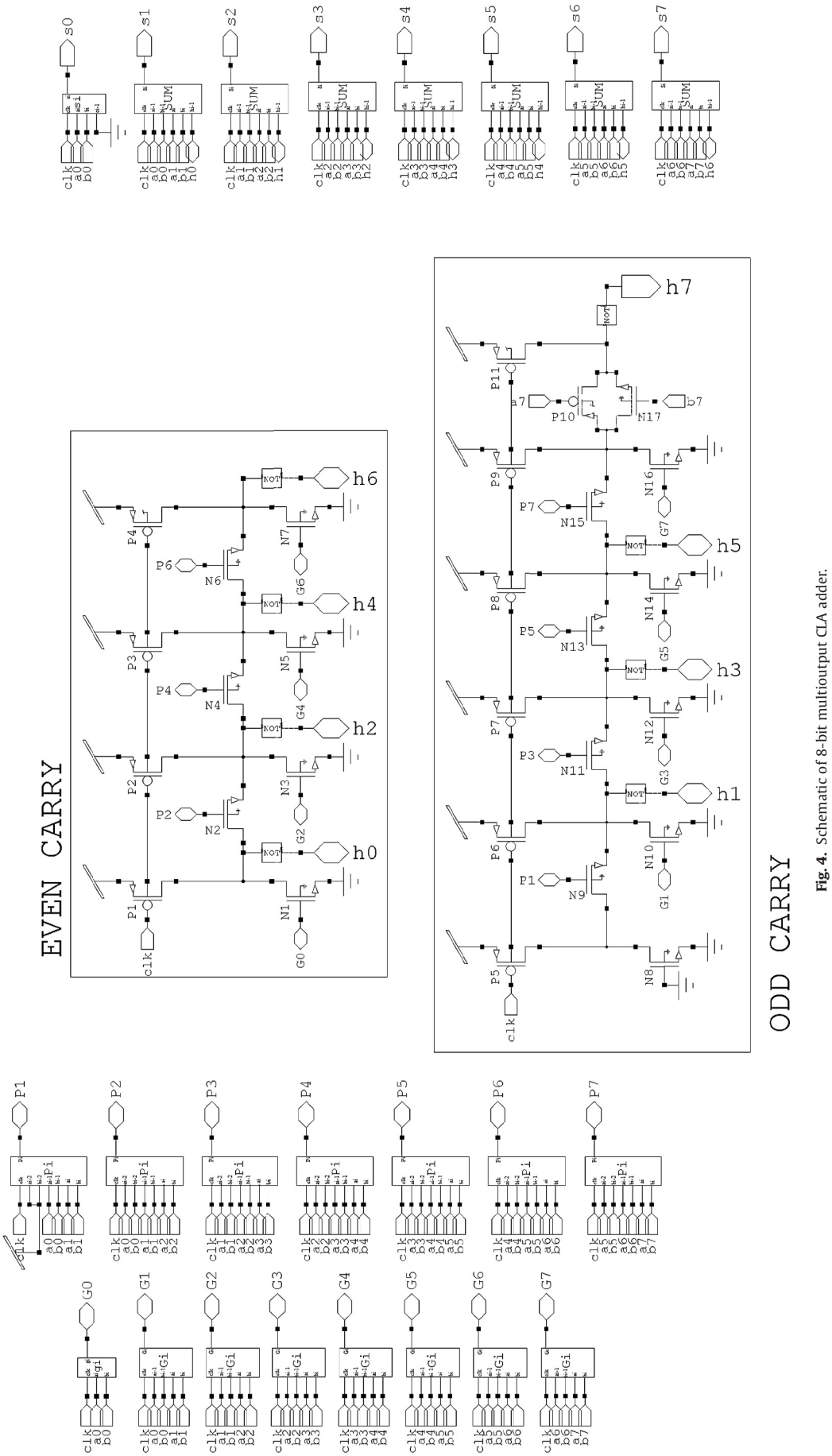

Fig. 4. Schematic of 8-bit multioutput CLA adder.

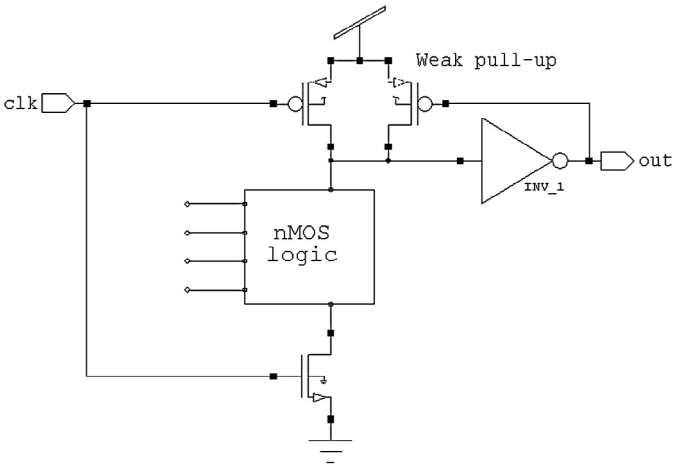

Fig. 5. Weak PMOS pull-up device in feedback loop [22].

logic are being proposed in the literature as demonstrated [18–20]. MCC is also implemented using static CMOS as demonstrated [21]. This results in improvement in terms of area-speed as compared to single-output gates. The main advantage of MCC adder is that it generates all carries in parallel using an iterative shared transistor structure and thus avoids the rippling of carries.

Domino CMOS logic is a type of Dynamic CMOS circuit. Domino CMOS are designed to allow unrestricted cascading of multiple stages in dynamic operation. The main goal is to achieve reliable, high-speed and compact circuits by using least complex clocking scheme as demonstrated [22]. Domino CMOS logic gates reduce number of transistors required to realize any Boolean function. For complex or large Boolean functions, there is a reduction in number of transistors in domino CMOS as compared to static CMOS logic. Cascading problem in dynamic CMOS stages is resolved in domino CMOS because all the input transistors in logic blocks are turned off during the pre-charge phase (clk = 0). During evaluation phase, each output can make a maximum of one transition (0 to 1), and thus each input of subsequent logic stages also makes at most one transition (0 to 1). The number of inverters in cascade must be even, so that during evaluation phase, the next domino CMOS stage inputs experience only 0 to 1 transition. Only non-inverting designs can be implemented using domino CMOS logic. If necessary, inversion must be applied using static CMOS logic. Charge sharing during evaluation phase results in erroneous outputs because charge sharing between the output node and the intermediate nodes of the NMOS logic results in decreasing the output voltage.

To prevent erroneous output due to charge sharing one solution is to add a weak PMOS transistor with small aspect ratio (W/L) to the dynamic CMOS stage output as shown in Fig. 5. This modified design forces high output logic unless there is a strong pull-down path between output node and ground. This modified circuit had been used to design High speed multioutput CLA Adder.

Efstathiou et al. [16] have recently shown the iterative carry formula for the CLA adder using MCC. Let A, B be two numbers having n-bits to add and S is the n-bit output sum of the addition.

The carry signal is represented by c_i and obtained by the recursive formula:

$$c_i = g_i + z_i \cdot c_{i-1} \quad (1)$$

where
$g_i = a_i.b_i$ is carry generate term
$z_i = t_i = a_i + b_i$ is carry propagate term
$z_i = p_i = a_i \oplus b_i$ is carry propagate term

The Domino implementation of generate (g_i), XOR propagate (p_i) and OR propagate (t_i) signals are shown in Fig. 6.

Expanding relation (1) gives

$$c_i = g_i + z_i g_{i-1} + z_i z_{i-1} g_{i-2} + \ldots + z_i z_{i-1} \ldots z_i g_0 + z_i z_{i-1} \ldots z_0 c_{-1} \quad (2)$$

where, c_{-1} is the input carry to the adder cell, $c_{-1} = 0$ for addition.

Efstathiou et al. [16] have recently shown the design of 8-bit carry lookahead adder composed of two independent carry chains. The two chains are known as *even carry chain* and *odd carry chain*. The carry chains compute eight carries required to design 8-bit CLA adder out of which four are even carries and other four are odd carries. A novel approach for high-speed parallel prefix Ling adders are proposed in Reference 23, in which the *even carries* h_0, h_2, h_4, h_6 are computed as given below.

From relation (1) we have $c_0 = g_0 + t_0 c_{-1}$, where c_{-1} is the input carry to the 8-bit CLA

$$c_0 = t_0.(g_0 + c_{-1}) = t_0 h_0 \text{ where, } h_0 = g_0 + c_{-1}$$

From relation (2) and taking $z_i = p_i$ with XOR gate

$$c_2 = t_2 (g_2 + g_1 + p_2 p_1 g_0 + p_2 p_1 p_0 c_{-1}) = t_2 (g_2 + g_1 + p_2 p_1 t_0 (g_0 + c_{-1})) = t_2 h_2$$
where $h_2 = g_2 + g_1 + p_2 p_1 t_0 (g_0 + c_{-1}) = g_2 + g_1 + p_2 p_1 t_0 h_0$

In the same way, h_4 and h_6 are derived as

$$h_4 = g_4 + g_3 + p_4 p_3 t_2 (g_2 + g_1 + p_2 p_1 t_0 (g_0 + c_{-1})) = g_4 + g_3 + p_4 p_3 t_2 h_2$$
$$h_6 = g_6 + g_5 + p_6 p_5 t_4 \ (g4 + g_3 + p_4 p_3 t_2 \ (g_2 + g_1 + p_2 p_1 t_0 \ (g_0 + c_{-1}))) = g_6 + g_5 + p_6 p_5 t_4 h_4$$

Similarly *odd carries* can be computed from relation (2) by putting i = 1, 3, 5, 7 to get h_1, h_3, h_5, h_7 respectively.

$$h_1 = g_1 + g_0 + p_1 p_0 c_{-1}$$
$$h_3 = g_3 + g_2 + p_3 p_2 t_1 h_1$$
$$h_5 = g_5 + g_4 + p_5 p_4 t_3 h_3$$
$$h_7 = g_7 + g_6 + p_7 p_6 t_4 h_5$$

Let, $Gi = g_i + g_{i-1}$ be new generate signal and $P_i = p_i\, p_{i-1}\, t_{i-2}$ be new propagate signal, where $g_{-1} = c_{-1}$ and $t_{-1} = 1$.

New signals G_i and P_i are similar to g_i and p_i as in (1), the only difference is that they use iterative method to produce the output by taking current and the previous bits for the operation. The new generate G_i and propagate P_i signals are shown in Fig. 6.

Vassiliadis [24] has shown that the sum bits are computed as follows

$$s_i = p_i \oplus c_{i-1} = p_i \oplus (t_{i-1} \cdot h_{i-1}) = p_i (t'_{i-1} + h'_{i-1}) + p'_i t_{i-1} h_{i-1} \quad \text{for } i > 0$$
$$s_i = h'_{i-1} \cdot p_i + h_{i-1} (p_i \oplus t_{i-1}) \quad (3)$$

$$s_0 = p_0 \oplus c_{-1}$$

3. Multiplier design

Multiplication is one of the most fundamental arithmetic operations; it finds application in Digital Signal processing. In this section 8-bit multiplier design is addressed. The multiplier is designed using the three adders used for partial product addition i.e., Full adder using Double Pass Transistor (DPL) and multioutput carry Lookahead logic (CLA).

The 8-bit multiplier design comprises a 4×4 multiplier and an 8-bit adder for partial product addition as shown in Fig. 7. A 4×4 array multiplier is designed using full adder cells and AND logic gates using static CMOS. The 4×4 array multiplier is shown in Fig. 8. Fig. 7 depicts the schematic of Full adder using Double Pass Transistor (DPL) and multioutput carry Lookahead logic (CLA) analyzed in this paper.

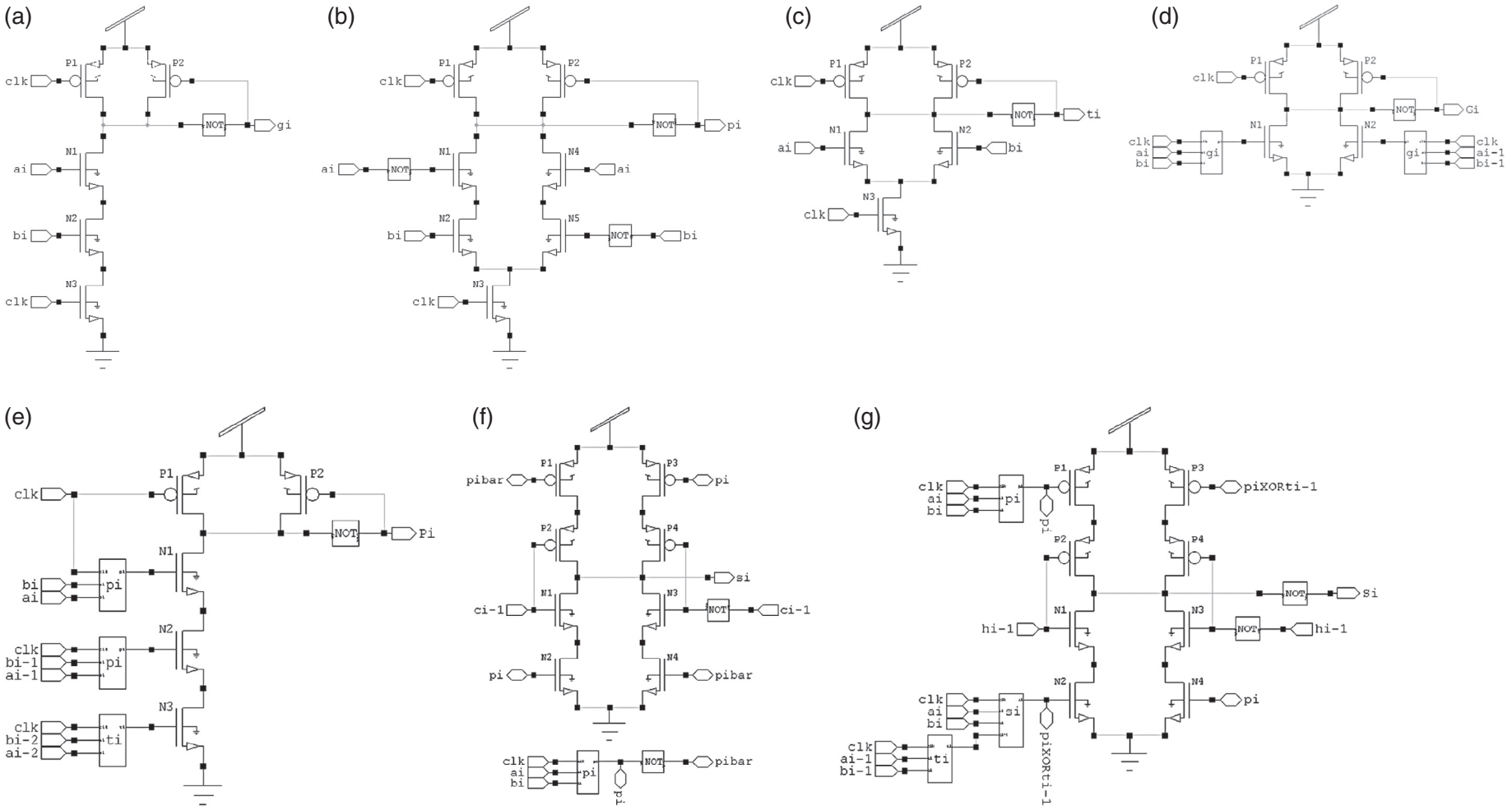

Fig. 6. Domino CMOS implementation [9] for (a) generate g_i; (b) propagate p_i; (c) propagate t_i; (d) New generate G_i; (e) New propagate P_i; (f) CMOS implementation for sum bit s_i; and (g) CMOS implementation for sum bit S_i.

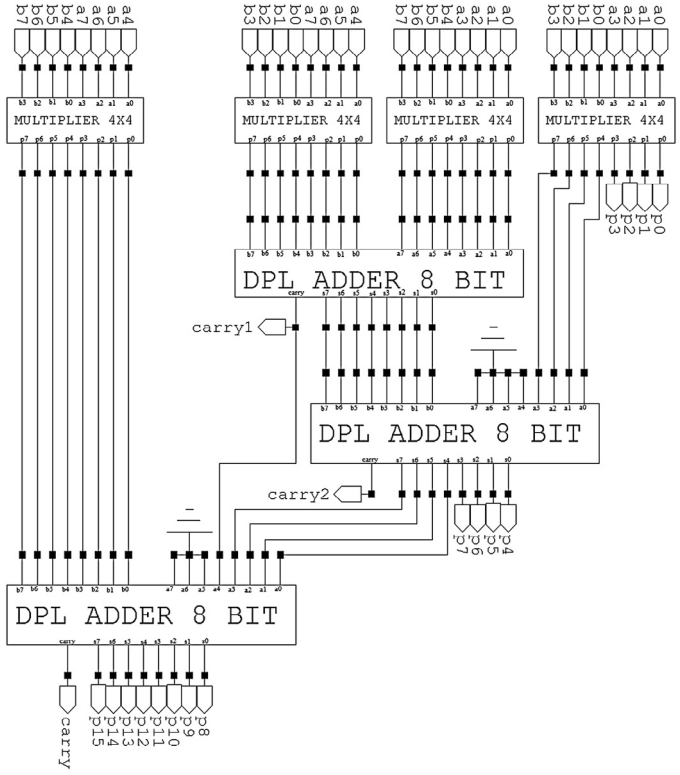

Fig. 7. Block diagram of an 8-bit multiplier.

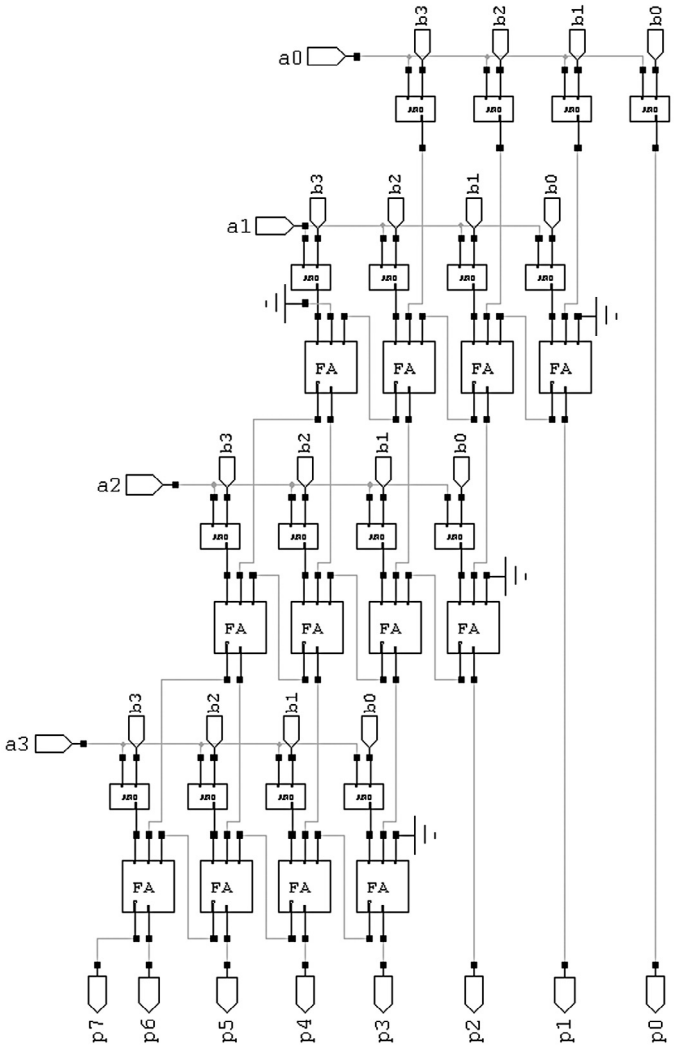

Fig. 8. Four bit multiplier design.

4. Simulation and results

In order to evaluate the comparative performance of the different 8-bit adders and multipliers, the circuits are implemented in 90 nm TSMC CMOS technology and simulated at 25 °C using HSpice (© Avant! Corporation). The simulation is done with different values of V_{DD} ranging from 0.6 V to 1.2 V. Each topology is analyzed in terms of propagation delay, power dissipation and their product. The propagation delay is measured as the time difference between the instant the input signal reaches 50% of its logic swing and the instant the output also reaches the same value.

The power dissipation is evaluated by estimating the power flowing into the circuit. For each value of supply voltage, we first performed the functional verification for each topology considering all possible input transitions. Comparison analysis of an 8-bit adder design and 8-bit multiplier design is depicted in Table 1 and plotted in Fig. 9.

It is apparent that multioutput Carry lookahead adder architecture has smaller delays, even though its speed advantage is greatly reduced for lower V_{DD}. At high supply voltages DPL and CLA adders are always faster than the earlier version. At $V_{DD} = 0.6$ V, propagation delay of multioutput CLA is 29.21% less than ripple carry adder and 6.09% less than DPL adder but it consumes more power in comparison with the other two adder designs. It is clear that the power dissipation for multioutput Carry lookahead adder architecture is always the highest. Hence, these new topologies should not be used when the primary target is low power consumption. Thus, multioutput CLA adder is more suitable for time critical operations.

Table 2 depicts the performance summary of 8-bit multiplier designs analyzed in this paper and their performance characteristics are plotted in Fig. 10.

Table 1
Performance analysis of 8-bit adders.

V_{DD} (volts)	Scheme	Delay (ns)	Power (μW)	Power delay product (fJ)
0.6	CMOS	3.478	16.75	58.256
	DPL	2.612	27.72	72.405
	CLA	2.462	101.88	250.828
0.7	CMOS	2.745	34.67	95.169
	DPL	1.681	51.21	86.089
	CLA	1.437	161.01	231.371
0.8	CMOS	1.984	62.44	123.881
	DPL	1.331	85.28	113.508
	CLA	1.096	370.42	405.98
0.9	CMOS	1.629	104.04	169.481
	DPL	1.085	138.74	150.533
	CLA	0.847	458.67	388.493
1	CMOS	1.395	138.74	193.542
	DPL	0.910	226.90	206.479
	CLA	0.769	714.54	549.481
1.1	CMOS	1.173	195.46	229.275
	DPL	0.814	314.16	255.726
	CLA	0.699	994.51	695.163
1.2	CMOS	1.091	252.16	275.106
	DPL	0.735	438.35	322.187
	CLA	0.661	1242.1	821.028

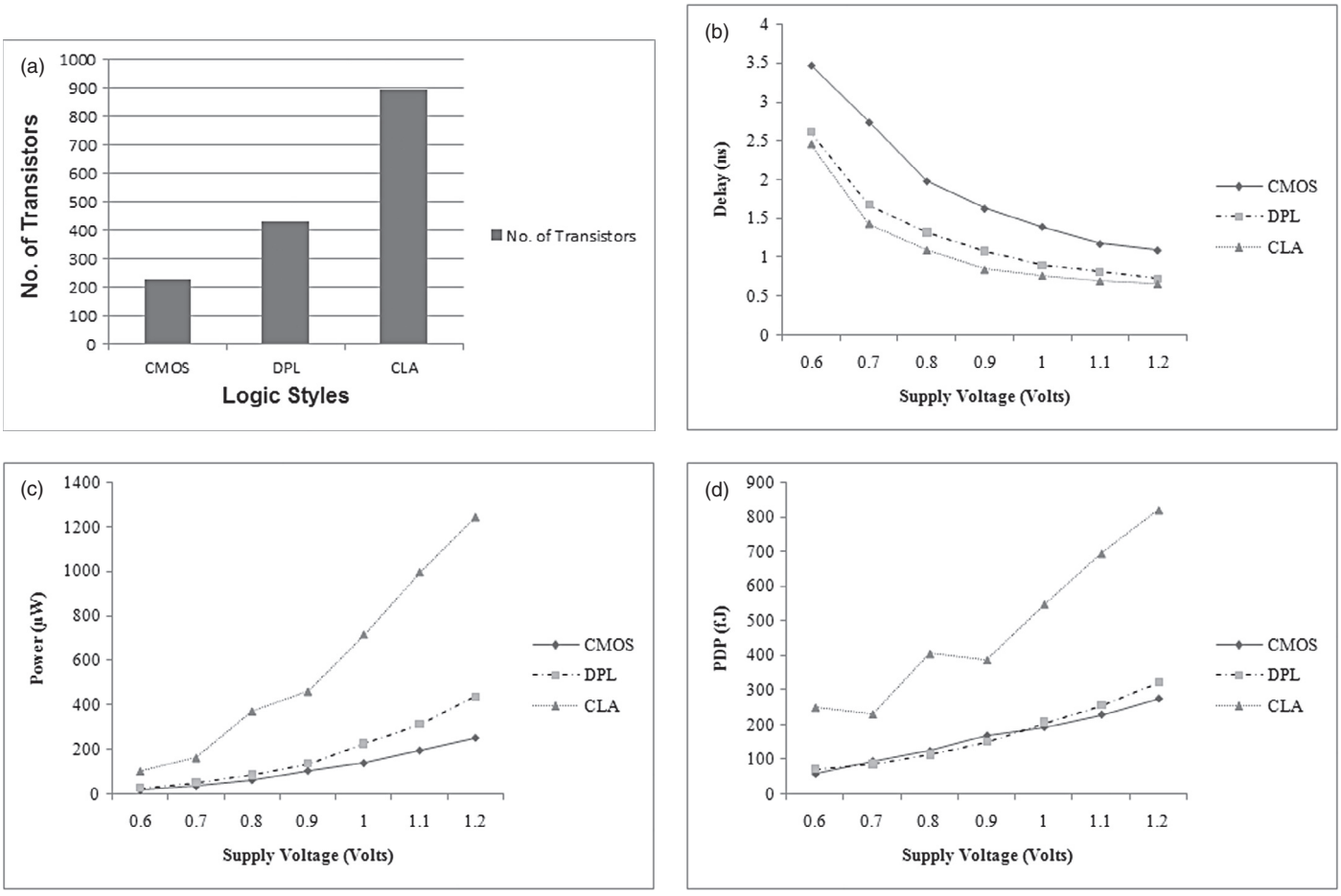

Fig. 9. Comparison analysis of adders on the basis of (a) Transistor Count, (b) Delay versus V_{DD}, (c) Power versus V_{DD}, and (d) Power Delay Product versus V_{DD}.

Table 2 depicts that multiplier topologies involving multioutput carry lookahead adder have the lowest delay as compared with other multiplier architectures. Eight-bit multiplier design using multioutput CLA adder has 26.39% less delay than the multiplier with CMOS full adder and 13% improvement in delay as compared to multiplier designed using DPL adder at $V_{DD} = 0.6$ V. Due to large number of transistors in CLA adder the power consumption of CLA adder is more than both CMOS and DPL adder structures. At $V_{DD} = 0.7$ V CLA multiplier power is 49.41% more than CMOS multiplier and 7.5% more than multiplier with DPL adder. On increasing the value of V_{DD} this gap increases. It is apparent that there is tradeoff between delay and power. Multiplier topologies involving multioutput carry lookahead adder consumes more power in comparison with the other topologies. As Multiplier topologies involving multioutput carry lookahead adder are power hungry, they are not suitable for low power applications.

Table 2
Performance summary of 8-bit multiplier.

V_{DD} (volts)	Scheme	Delay (ns)	Power (mW)	Power delay product (pJ)
0.6	CMOS	7.913	0.172	1.361
	DPL	6.696	0.239	1.600
	CLA	5.825	0.257	1.497
0.7	CMOS	5.172	0.384	1.986
	DPL	4.631	0.469	2.172
	CLA	4.460	0.540	2.408
0.8	CMOS	3.862	0.723	2.792
	DPL	3.451	0.753	2.599
	CLA	3.274	0.837	2.740
0.9	CMOS	3.086	1.064	3.283
	DPL	2.782	1.259	3.503
	CLA	2.569	1.204	3.093
1	CMOS	2.588	1.672	4.327
	DPL	2.375	1.974	4.688
	CLA	2.201	1.701	3.744
1.1	CMOS	2.258	2.259	5.109
	DPL	2.102	2.972	6.247
	CLA	1.963	2.375	4.662
1.2	CMOS	2.019	2.885	5.825
	DPL	1.924	4.016	7.727
	CLA	1.775	3.175	5.636

5. Conclusion

In this paper we have proposed speed efficient multiplier architecture designed using multioutput carry lookahead adder. Further, we have carried out a comparison among the multipliers employing latest adder architectures. The comparison results are obtained in power-delay space. Thus, the design guidelines are derived to make the selection of appropriate multiplier design at the beginning of the design process. At $V_{DD} = 0.6$ V the 8-bit multiplier designed using multioutput CLA adder has 26.39% less delay than the multiplier with CMOS full adder and 13% improvement in delay as compared to multiplier designed using DPL adder. Multiplier topologies involving multioutput carry lookahead adder architecture are competitive in terms of power and power delay product overheads but it proves to be more speed efficient. This new version is more suitable for high speed applications.

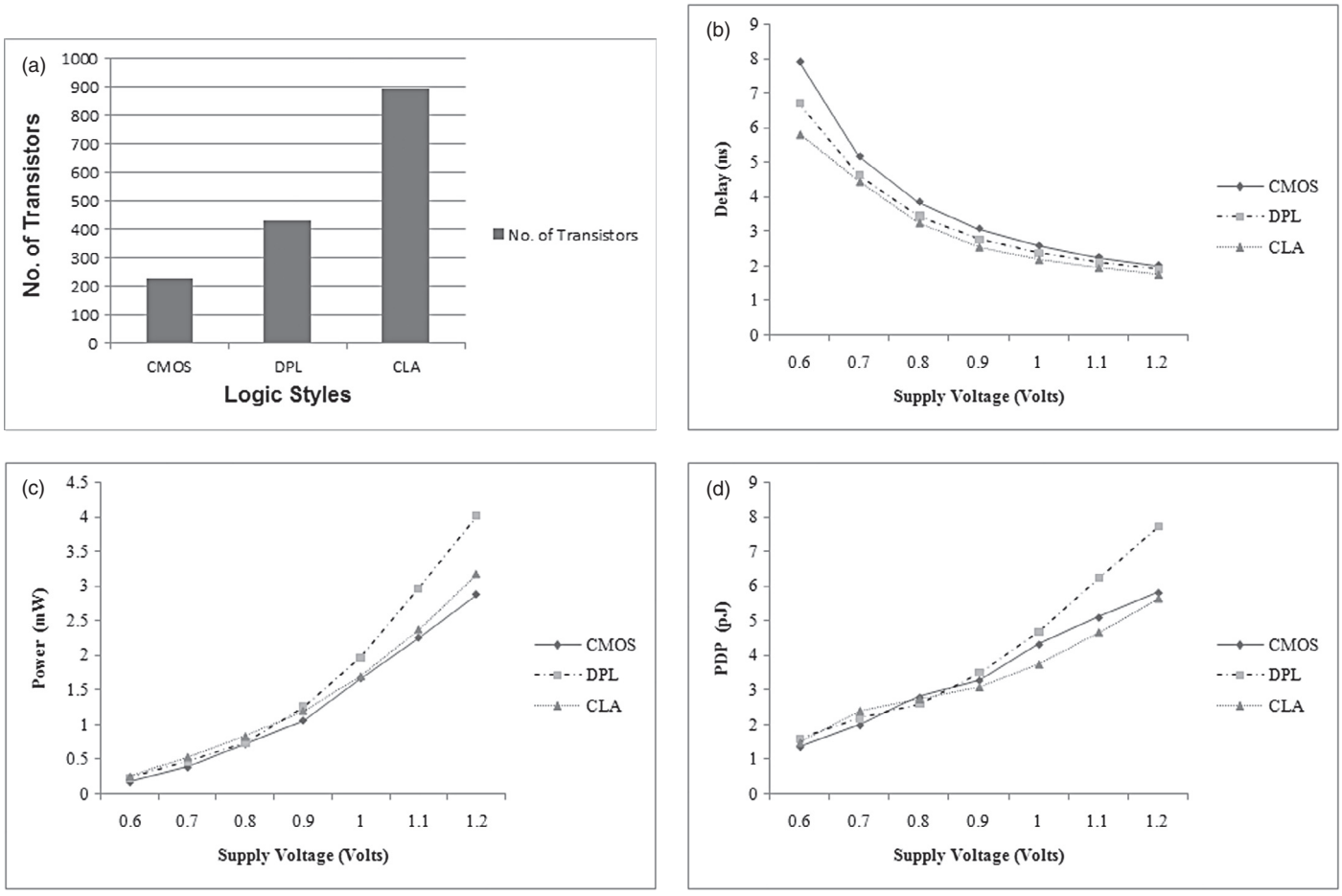

Fig. 10. Comparison analysis of multipliers on the basis of (a) Transistor Count, (b) Delay versus V_{DD}, (c) Power versus V_{DD}, and (d) Power Delay Product versus V_{DD}.

References

[1] K. Hwang, Computer Arithmetic: Principles, Architecture, and Design, Wiley, New York, NY, USA, 1979.

[2] Y.-J. Jang, Y. Shin, M.-C. Hong, J.-K. Wee, S. Lee, Low-Power 32bit×32bit Multiplier Design with Pipelined Block-Wise Shutdown: High Performance Computing – HiPC 2005, 12th International Conference, Goa, India, Springer, 3769, pp. 398–406, 2005.

[3] Y. Liu, S. Furber, The Design of an Asynchronous Carry-Lookahead Adder Based on Data Characteristics: Integrated Circuit and System Design. Power and Timing Modeling, Optimization and Simulation, 15th International Workshop, PATMOS 2005, Leuven, Belgium, Springer, 3728, pp. 647–656, 2005.

[4] K. Tiwari, A. Khopade, P. Jadhav, Optimized Carry Look-Ahead BCD Adder Using Reversible Logic: Technology Systems and Management Communications in Computer and Information Science, First International Conference, ICTSM 2011, Mumbai, India, Springer, 45, pp. 260–265, 2011.

[5] A. Weinberger, J.L. Smith, A logic for high speed addition, Nat. Bureau Stand. Circulation 591 (1958) 3–12.

[6] O. Kavehei, M. Rahimi Azghadi, K. Navi, A.P. Mirbaha, Design of robust and high-performance 1-bit CMOS Full Adder for nanometer design, IEEE Comput. Soc. Annu. Symp. VLSI ISVLSI 08 Montpellier Fr., pp. 10–15, 2008.

[7] M. Haghparast, K. Navi, A novel reversible BCD adder for nanotechnology based systems, Am. J. Applied Sci. 5 (3) (2008) 282–288 ISSN 1546-9239.

[8] Z. Huang, High-level optimization techniques for low-power multiplier design (Ph.D. thesis), University of California, 2003.

[9] L. Junming, S. Yan, L. Zhenghui, W. Ling, A novel 10-transistor low-power high-speed full adder cell, in: 6th International Conference on Solid-State and Integrated-Circuit Technology, 2001, pp. 1155–1158.

[10] Y. Jiang, A. Al-Sheraidah, Y. Wang, E. Sha, J.-G. Chung, A novel multiplexer-based low-power full adder, IEEE Trans. Circ. Syst. II: Exp. Briefs 51 (2004) 345–348.

[11] C.-H. Chang, M. Zhang, J. Gu, A novel low power low voltage full adder cell, in: 3rd International Symposium on Image and Signal Processing and Analysis, IEEE, New York, 2003, pp. 454–458.

[12] N. Weste, D. Harris, CMOS VLSI Design, A Circuit and System Perspective, Addison-Wesley, USA, 2004.

[13] M. Ito, D. Chinnery, K. Keutzer, Low power multiplication algorithm for switching activity reduction through operand decomposition, in: 21st International Conference on Computer Design, 2003, pp. 21–26.

[14] A. Kishore Kumar, D. Somasundareswari, V. Duraisamy, T. ShunbagaPradeepa, Design of Low Power Multiplier with Energy Efficient Full Adder Using DPTAAL, Hindawi Publishing Corporation, 2013, doi:10.1155/2013/157872.

[15] R. Zimmermann, W. Fichtner, Low-power logic styles: CMOS versus pass-transistor logic, IEEE J. Solid-St. Circ. 32 (7) (1997) 1079–1090.

[16] C. Efstathiou, Z. Owda, Y. Tsiatouhas, New high-speed multioutput carry look-ahead adders, IEEE Trans. Circ. Syst. II 60 (10) (2013) 667–671.

[17] M. Morris Mano, M.D. Ciletti, Digital Design, fourth ed., Pearson, 2008.

[18] P.K. Chan, M.D.F. Schlag, Analysis and design of CMOS Manchester adders with variable carry-skip, IEEE Trans. Computers 39 (8) (1990) 983–992, Aug. 1990.

[19] S. Perri, P. Corsonello, F. Pezzimenti, V. Kantabutra, Fast and energy-efficient Manchester carry-bypass adders, IEEE Proc. Circ. Dev. Syst. 151 (6) (2004) 497–502.

[20] Z. Wang, G. Jullien, W. Miller, J. Wang, S. Bizzan, Fast adders using enhanced multiple-output domino logic, IEEE J. Solid-St. Circ. 32 (2) (1997) 206–214.

[21] G.A. Ruiz, New static multi-output carry look-ahead CMOS adders, IEEE Proc. Circ. Dev. Syst. 144 (6) (1997) 350–354.

[22] S.-M. Kang, Y. Leblebici, CMOS Digital Integrated Circuits Analysis and Design, third ed., Tata McGraw-Hill, 2003.

[23] G. Dimitrakopoulos, D. Nikolos, High-speed parallel-prefix VLSI Ling adders, IEEE Trans. Computers 54 (2) (2005) 225–231.

[24] S. Vassiliadis, Recursive equations for hardwired binary adders, Int. J. Elect. 67 (2) (1989) 201–213, Aug. 1989.